

U.S. PATENT APPLICATION

**Title: RESOURCE ARBITRATION IN ACCORDANCE WITH A
 MASKED REQUEST VECTOR**

Inventor(s): Sreenath Kurupati

Filing Date: June 23, 2003

Docket No.: P16515

**Prepared by: Patrick Buckley
 Buckley, Maschoff, Talwalkar & Allison LLC
 Five Elm Street
 New Canaan, CT 06840
 (203) 972-0191**

RESOURCE ARBITRATION IN ACCORDANCE WITH A MASKED REQUEST VECTOR

BACKGROUND

A number of requestors may want to use a limited resource. For example, a number of requestors associated with different input ports may receive information packets that need to be processed and transmitted via a single output port. A resource allocation unit may then allocate the resource to one of those requestors.

The resource allocation unit could scan a list of requestors (starting with the first requestor in the list) and simply allocate the resource to the first requestor that is currently requesting the resource. This approach, however, may not be fair. If, for example, the first requestor in the list constantly requests the resource, the resource will never be allocated to other requestors.

To avoid this, a "round-robin" approach to resource allocation may be used. For example, a resource allocation unit might scan a list of requestors and allocate the resource to the first requestor that is currently requesting the resource. The next time the resource is to be allocated, however, the resource allocation scans the list starting with the next requestor (*e.g.*, if the resource was allocated to the fourth requestor during one cycle, the list would be scanned starting with fifth requestor during the next clock cycle).

A round-robin approach, however, may be difficult to implement. For example, the hardware used to implement the approach might consume a significant amount of silicon area (*e.g.*, variable shift logic may result in a high gate count, especially when there are a large number of requestors). Moreover, it may take a significant amount of time to perform the resource allocation (*e.g.*, it may be difficult to perform the allocation in a single clock cycle).

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 is a method according to some embodiments.

FIG. 2 illustrates a masked request vector according to some embodiments.

FIG. 3 is a resource allocation unit according to some embodiments.

5 FIG. 4 is an example of a resource allocation system according to some
embodiments.

FIG. 5 is a resource allocation method according to some embodiments.

FIG. 6 is a switch according to some embodiments.

FIG. 7 illustrates a masked request vector according to another embodiment.

10 DETAILED DESCRIPTION

Some embodiments described here are associated with "requestors." As used herein, the term "requestor" can apply to any function unit (*e.g.*, a hardware or software unit) that might need to access a resource. By way of example only, consider an Ethernet switch or router that operates in accordance with the Fast Ethernet Local Area Network
15 (LAN) transmission standard 802.3-2002® published by the Institute of Electrical and
Electronics Engineers (IEEE). The switch or router may have a number of input ports
(*e.g.*, 27 input ports), and information packets may be received at several of those input
ports. In this case, a Media Application Control (MAC) module associated with each
input port might act as a requestor that will ask to have an information packet processed
20 by a resource. A packet parser unit may then select the requestor to which the resource
will be allocated (and thus which information packet will be processed).

Method

FIG. 1 is a flow chart of a method according to some embodiments. The flow charts described herein do not necessarily imply a fixed order to the actions, and embodiments may be performed in any order that is practicable. Note that any of the methods described herein may be performed by firmware, hardware, software, or any combination of these techniques. For example, a storage medium may store thereon instructions that when executed by a machine result in performance according to any of the embodiments described herein.

At 102, a request vector is determined, and each bit in the request vector represents a requestor. A portion of the request vector is masked at 104 based on a previously selected requestor (*e.g.*, the requestor to which the resource was allocated during the prior clock cycle). At 106, a requestor is selected in accordance with the masked request vector.

Consider, for example, the N requestors 210 illustrated in FIG. 2 along with the associated N bits in the request vector 220 (*i.e.*, bits $[b_{N-1}, \dots, b_0]$). In this example, b_L is associated with the last requestor that received the resource.

According to this embodiment, a bit with a value of one indicates that the associated requestor is currently requesting a resource, and a value of zero indicates that the requestor is not currently requesting the resource (*i.e.*, at least bits b_{N-1} , b_{L+2} , and b_1 are currently requesting the resource in FIG. 2). According to other embodiments, of course, a zero could indicate that a requestor is asking for a resource while a one indicates that it is not.

A masked request vector 230 is created by masking bits b_L through b_0 in the request vector 220 (*e.g.*, bits b_L through b_0 may be set to zero as illustrated by the shaded bits in FIG. 2).

A requestor 210 is then selected by starting with b_0 and looking to find the first bit with a value of one in the masked request vector 230 (*e.g.*, the least significant bit that

has a value of one). In this example, b_{L+2} might be selected (note that b_1 is not selected because it is masked out in the masked request vector 230). The resource can then be allocated to requestor $L+2$. The next time the resource is allocated (*e.g.*, during the next clock cycle), bits b_{L+2} through b_0 will be masked out in the masked request vector 230
5 (because at that point, b_{L+2} will be associated with the last requestor that received the resource).

Resource Allocation Unit

FIG. 3 is a resource allocation unit 300 according to some embodiments. The resource allocation unit 300 includes a masking unit 310 that receives a request vector
10 (*e.g.*, having bits $[b_{N-1}, \dots, b_0]$) along with a signal indicating the last selected requestor (*e.g.*, L). The masking unit 310 generates a masked request vector (*e.g.*, by masking bits b_L through b_0).

A priority encoder 320 receives the masked request vector from the masking unit 310. The priority encoder 320 outputs a signal indicating a selected requestor (*e.g.*, by
15 finding the least significant bit that has a value of one in the masked request vector). Since the masking unit 310 and the priority encoder 320 are used to select a requestor, a round-robin allocation of resources may be efficiently implemented (*e.g.*, in silicon). Moreover, the allocation may be performed more quickly as compared to an implementation that uses variable shift logic.

20 Example

FIG. 4 is an example of a resource allocation system 400 according to some embodiments. As before, each of N requestors 410 provide a request signal (*e.g.*, a one or zero) to establish a request vector 420 having bits $[b_{N-1}, \dots, b_0]$. This request vector 420 is provided to a masking unit 430.

The masking unit 430 also receives an indication of the previously selected requestor (L). The masking unit 430 may then generate an N -bit mask vector having bits $[m_{N-1}, \dots, m_0]$, where m_i is set to one if $i > L$ and to zero if $i \leq L$.

5 The masking unit 430 may generate a masked request vector by combining the request vector with the mask vector via a Boolean AND operation. In the masked request vector, therefore, bits b_{N-1} through b_{L+1} will be the same as they were in the request vector, and bits b_L through b_0 will be set to zero.

10 The masked request vector is provided from the masking unit 430 to a first priority encoder 440. The first priority encoder 440 may then generate a masked selection signal (having $\log(N)$ bits) reflecting the least significant bit in the masked request vector that indicates the requestor is requesting a resource. If an eight-bit masked request vector was $[0, 1, 0, 0, 1, 0, 0, 0]$, for example, then the masked selection signal would be $[0, 1, 1]$ representing "three" (because bit b_3 was the least significant bit that had a value of one).

15 As another example, consider a system with six requestors. Moreover, assume that the resource was allocated to the fifth requestor during the last clock cycle and that the current request vector is $[0, 0, 1, 0, 1, 0]$. In this case, the masked request vector will be $[0, 0, 0, 0, 0, 0]$ - and the first priority encoder 400 will not find any bit in the masked request vector that indicates the requestor is requesting a resource. Not selecting any
20 requestor in this situation, however, would be inefficient because two requestors actually want to use the resource (*i.e.*, the second and fourth requestors).

To avoid this result, a second priority encoder 450 may receive the original (un-masked) request vector 420 and generate an un-masked selection signal (again having $\log(N)$ bits) representing the least significant bit in request vector 420 that indicates a
25 requestor is requesting a resource. Note that the first and second priority encoders 440, 450 may operate in parallel.

A selector unit 460 may receive the masked selection from the first priority encoder 440 and the un-masked selection from the second priority encoder 450. In

addition, according to some embodiments, the selector unit 460 receives a present signal (e.g., a one or a zero) from each of the first and second priority encoders 440, 450 indicating whether or not least one of the bits in the respective vectors had a value of one.

The selector unit 460 then generates a selected requestor signal as follows. If the present signal from the first priority encoder 440 has a value of one (indicating that at least one bit in the masked request vector had a value of one), then the selected requestor signal is the masked selection signal received from the first priority encoder 440. If present signal from the first priority encoder is zero (indicating that all of the bits in the masked request vector were zero), then the selected requestor signal is the un-masked selection signal received from the second priority encoder 450.

The selected requestor may be provided to the masking unit 430 (so that the masking unit 430 will know which bits should be masked during the next clock cycle) and to a grant generator 470. According to some embodiments, the grant generator 470 is a decoder that generates an N-bit grant vector, with each bit being provided to the appropriate requestor 410 (e.g., a single bit in the grant vector may have a value of one to indicate that requestor that will be given access to the resource).

The resource allocation portion of the system 400 (e.g., elements other than the requestors 410) may have a gate count of approximately two when implemented in silicon. Moreover, the timing delay may be similar to a single priority encoder because the two priority encoders are implemented in parallel.

The resource allocation system 400 may be associated with, for example, a packet network, a LAN, an Ethernet network, a switch, and/or a router. Moreover, elements of the system may be implemented as an Application Specific Integrated Circuit (ASIC) device, a Field-Programmable Gate Array (FPGA) device, and/or a custom integrated circuit.

FIG. 5 is a resource allocation method according to some embodiments. The method may be performed, for example, in accordance with the resource allocation

system 400 of FIG. 4. At 502, a request vector is determined (*e.g.*, based on information received from a number of different requestors).

A masked request vector is then created at 504 based on the request vector and the last selected requestor. A priority encoder receives and processes the masked request
5 vector at 506. While the masked request vector is created and/or processed, another priority encoder process the un-masked request vector at 508.

If there is masked selection present at 510 (*e.g.*, at least one bit in the masked request vector has a value of one), the resource is allocated to a requestor based on the masked selection at 512. If there is no masked selection present at 510 (*e.g.*, all bits in
10 the masked request vector have a value of zero), the resource is allocated to a requestor based on the un-masked selection at 514. The process then continues at 502.

Switch

FIG. 6 is a switch 600 according to some embodiments. The switch 600 may be, for example, a 24+2 Gigabit Ethernet device. The switch 600 includes a resource
15 allocation unit 610 that operates in accordance with any of the embodiments described herein. The switch 600 also includes an Ethernet interface 620 that may facilitate an exchange of information via a network. According to other embodiments, the switch 600 includes an Asynchronous Transfer Mode (ATM) interface instead of, or in addition to, the Ethernet interface 620.

20 Additional Embodiments

The following illustrates various additional embodiments. These do not constitute a definition of all possible embodiments, and those skilled in the art will understand that many other embodiments are possible. Further, although the following embodiments are briefly described for clarity, those skilled in the art will understand how to make any

changes, if necessary, to the above description to accommodate these and other embodiments and applications.

In some of the embodiments described herein, a resource allocation unit provides an output that indicates a selected requestor. In other embodiments, however, different
5 information associated with a selected requestor could be provided. For example, a resource allocation unit might output an information packet associated with a selected requestor.

In addition, some embodiments are directed to networks (*e.g.*, packet networks, LANs and/or Ethernet networks) and network devices (*e.g.*, switches and/or routers).
10 Note, however, that embodiments may also be directed to other types of resource allocation and/or arbitration (*e.g.*, outside of the network context).

Similarly, in some embodiments a resource allocation unit selects a single requestor. According to other embodiments, the unit can instead select two or more requestors (*e.g.*, a resource might be able to handle two requestors at the same time).

15 Although specific approaches have been described herein, any number of other approaches may be implemented. Consider, for example, FIG. 7 which illustrates a masked request vector according to another embodiment. As before, N requestors 710 are associated with N bits in a request vector 720 (*i.e.*, bits $[b_{N-1}, \dots, b_0]$), and b_L is associated with the last requestor that received a resource. According to this approach, a
20 masked request vector 730 is created by masking bits b_{N-1} through b_L in the request vector 720. A requestor 710 is then selected by finding the most significant bit (as opposed to the least significant bit) that has a value of one in the masked request vector 730. In this example, b_1 might be selected (note that b_{N-1} is not selected because it is masked out in the masked request vector 730). The resource can then be allocated to
25 requestor 1.

The several embodiments described herein are solely for the purpose of illustration. Persons skilled in the art will recognize that other embodiments may be practiced with modifications and alterations limited only by the claims.